Data Structures I : Implementation of lists

UNIVERSIDAD EAFIT

Mauricio Toro
Department of Systems and Informatics
Universidad EAFIT

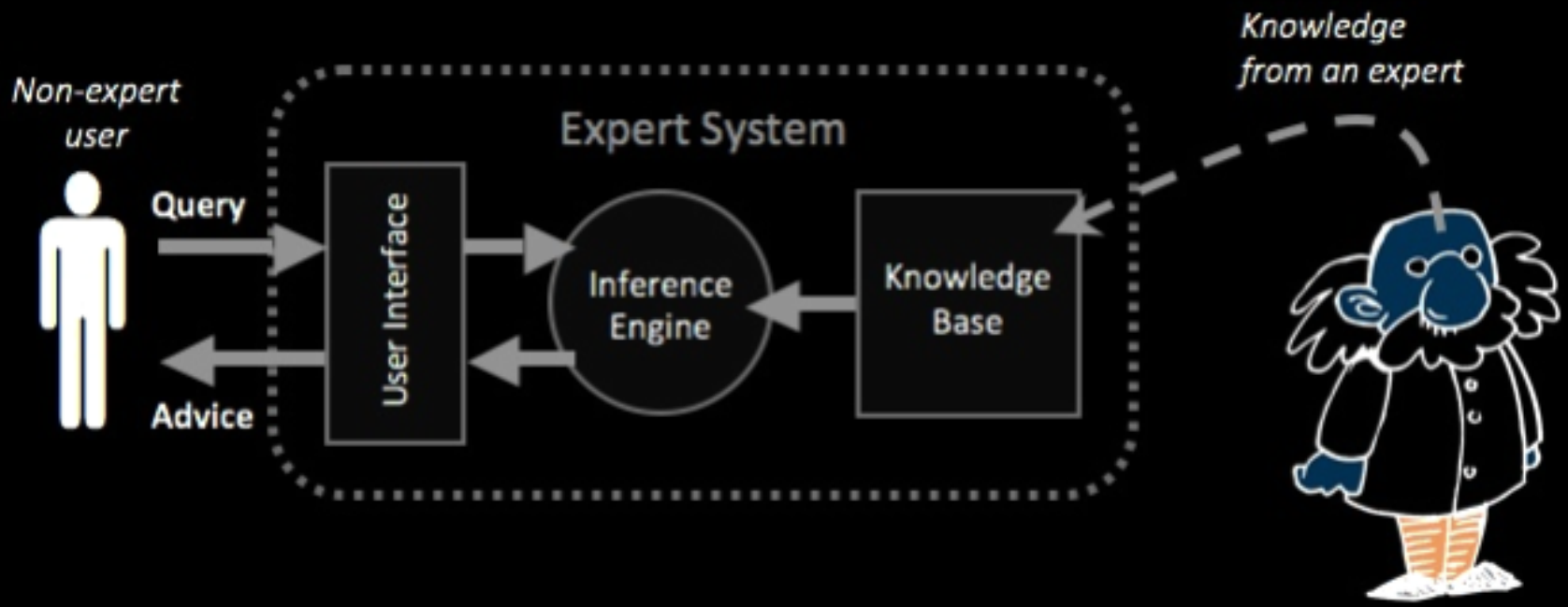
Cocktail of the day: Cosmopolitan

Disclaimer: Keep alcohol out of the hands of minors.

Cocktail of the day: Cosmopolitan

- 40 ml Vodka
- 15 ml Cointreau
- 15 ml Lime juice
- 30 ml Cranberry juice

Expert systems use trees



https://www.youtube.com/watch?v=uWEahgy3Iyc

Review: Lists

- Insertion in an array is slow ($O(n)$); insertion in a linked list is fast ($O(1)$).
- Random access in an array is fast ($O(1)$); random access in a linked list is slow ($O(n)$).
- Backward traversal of Singly-linked lists is slow ($O(n^2)$); for Doubly-linked lists and arrays is fast ($O(n)$).

Review: Lists

- Insertion in an array is slow ($O(n)$); insertion in a linked list is fast ($O(1)$).
- Random access in an array is fast ($O(1)$); random access in a linked list is slow ($O(n)$).
- Backward traversal of Singly-linked lists is slow ($O(n^2)$); for Doubly-linked lists and arrays is fast ($O(n)$).

Review: Lists

- Insertion in an array is slow ($O(n)$); insertion in a linked list is fast ($O(1)$).
- Random access in an array is fast ($O(1)$); random access in a linked list is slow ($O(n)$).
- Backward traversal of Singly-linked lists is slow ($O(n^2)$; for Doubly-linked lists and arrays is fast ($O(n)$).

Beware of references!

Binky pointer fun!
`http://www.cs.stanford.edu/cslibrary/PointerFunJavaBig.avi`

Implementation of Links for Singly-linked lists

```
class Link {
  public int data; //data
  public Link next; //ref. to next link
  public Link(int data); //constructor
}
```

Taken from [Lof98].

Implementation of Singly-linked lists

```
class LinkedList
{
private Link first;
public void LinkedList(); // constructor
public void insertFirst(int data);
...
}
```

Taken from [Laf98].

Methods for class Singly-linked List

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

Taken from [Laf98].

Methods for class Singly-linked List

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

Taken from [Laf98].

Methods for class Singly-linked List

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

Taken from [Laf98].

Methods for class Singly-linked List

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

Taken from [Laf98].

UNIVERSIDAD EAFIT

Methods for class Singly-linked List

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

Taken from [Laf98].

Methods for class Singly-linked List

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

Taken from [Laf98].

Methods for class Singly-linked List

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

Taken from [Laf98].

Linked lists

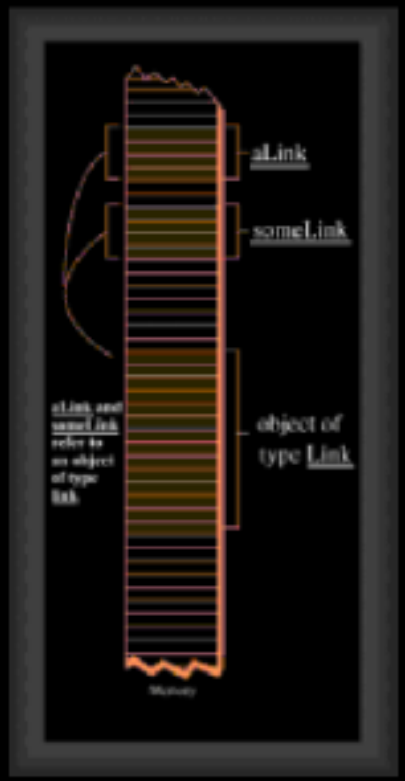


Figure: Links and references in memory. Taken from [Laf98].

Insert a new link

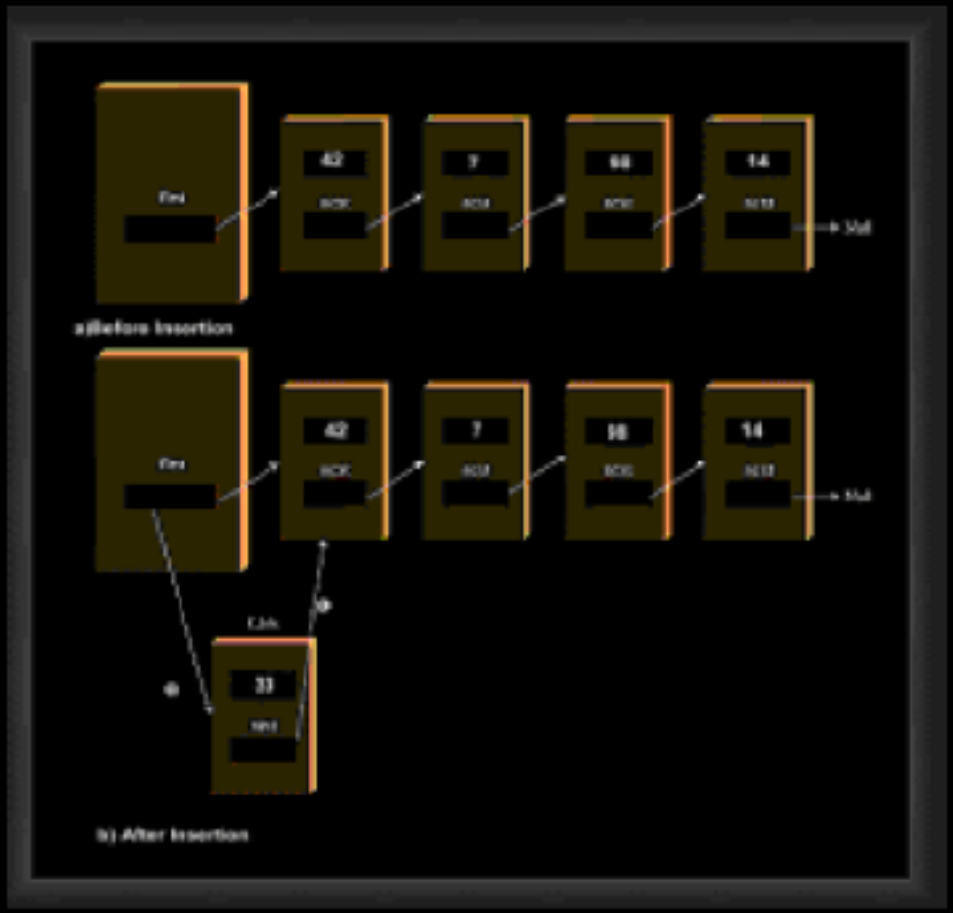


Figure: Inserting a new link. Taken from [Laf98].

Delete a link

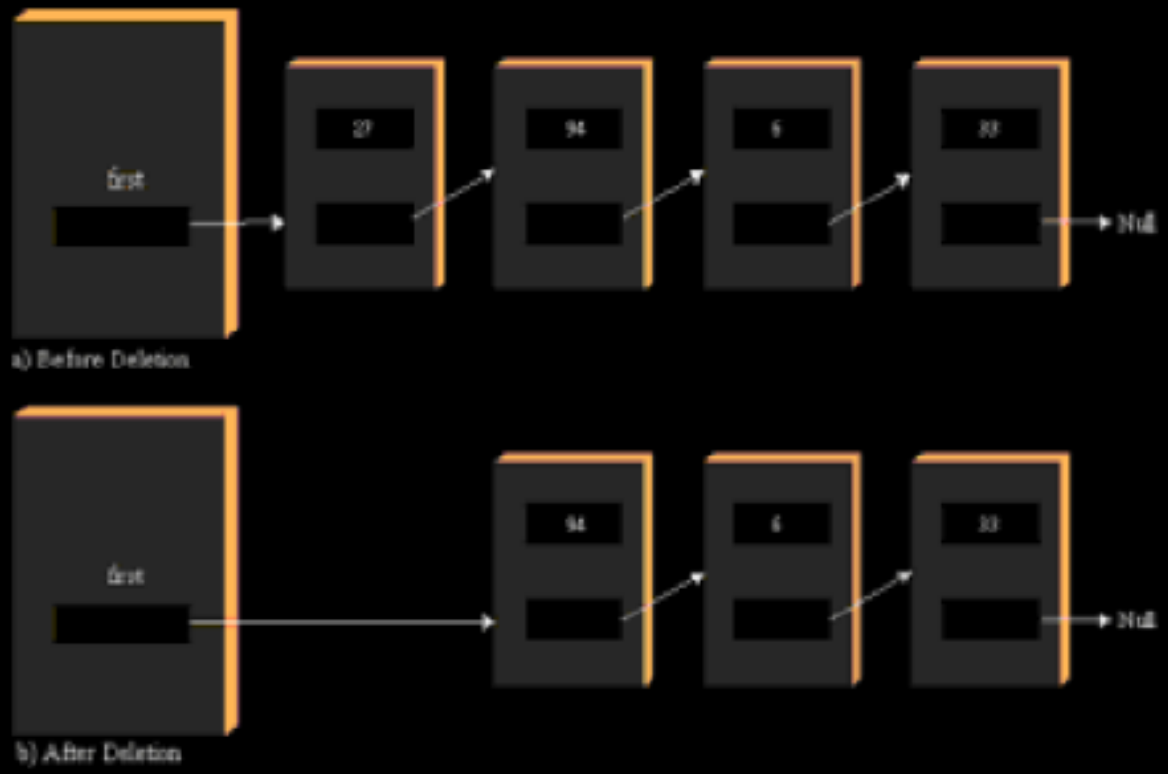


Figure: Deleting a link. Taken from [Laf98].

Simulator of a Linked list

http://visualgo.net/list.html

Methods for class Singly-linked List

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

Taken from [Laf98].

Implementation of Array Lists

```
class ArrayList
{
private int index;
private int[] data;
public void ArrayList(); // constructor
public void insertFirst(int data);
...
}
```

Methods for class ArrayList

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

Taken from [Laf98].

ArrayList

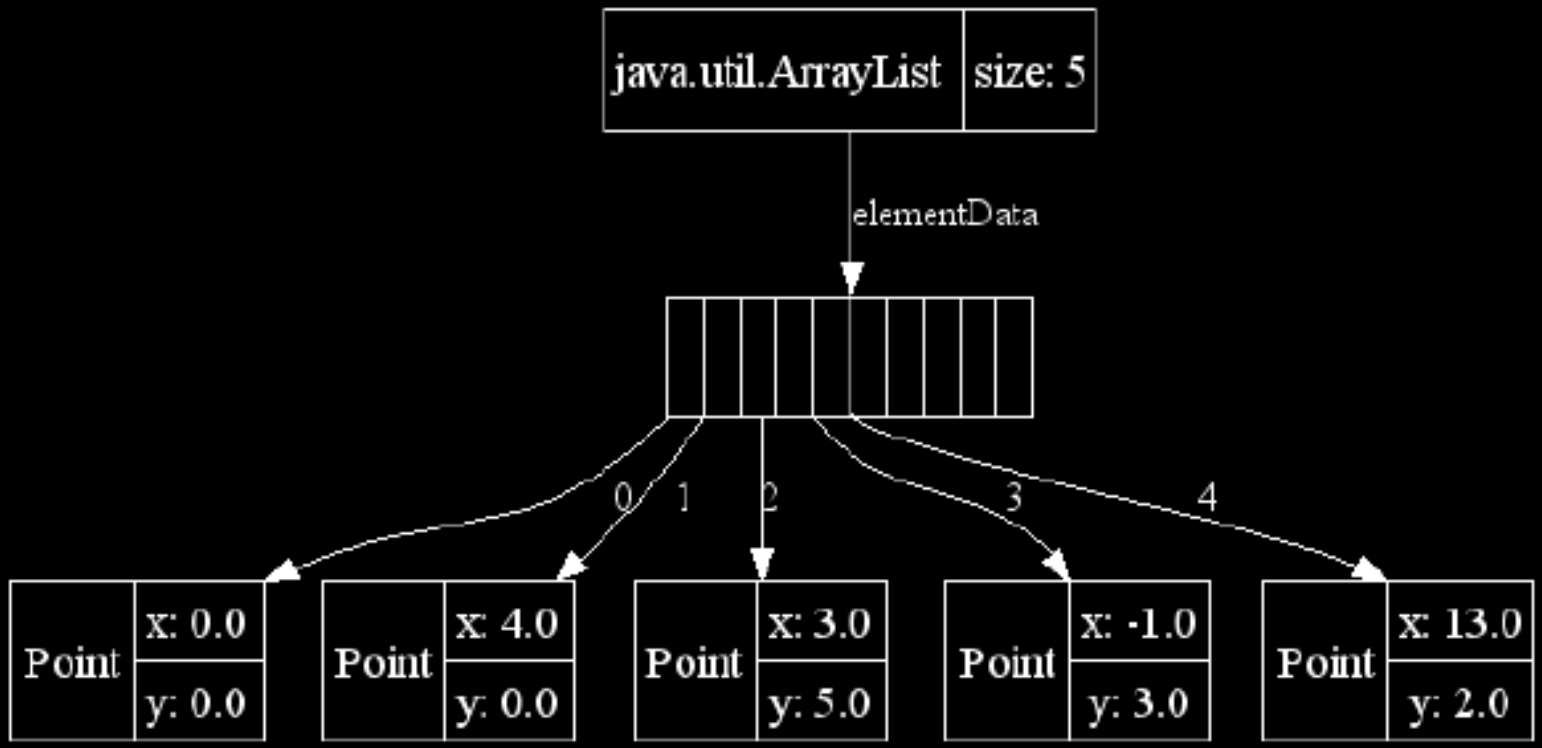


Figure: Representation of an ArrayList. Taken from `http://2.bp.blogspot.com/`

ArrayList's operations

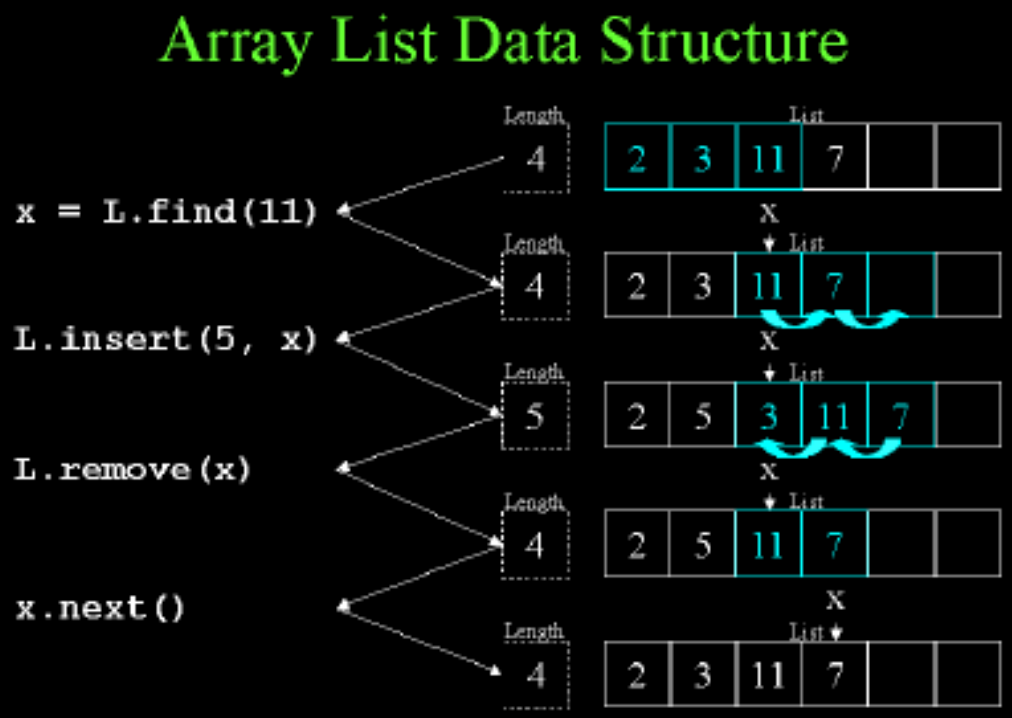


Figure: ArrayList's operations. Taken from `http://courses.cs.washington.edu/`.

Applet to simulate an Array list

http://www.cs.armstrong.edu/liang/animation/web/ArrayList.html

Methods for class ArrayList

1. int size()
2. void insertFirst(int data)
3. void deleteFirst()
4. void deleteLast()
5. void insertLast(int data)
6. boolean contains(int data)
7. int get(int index)...

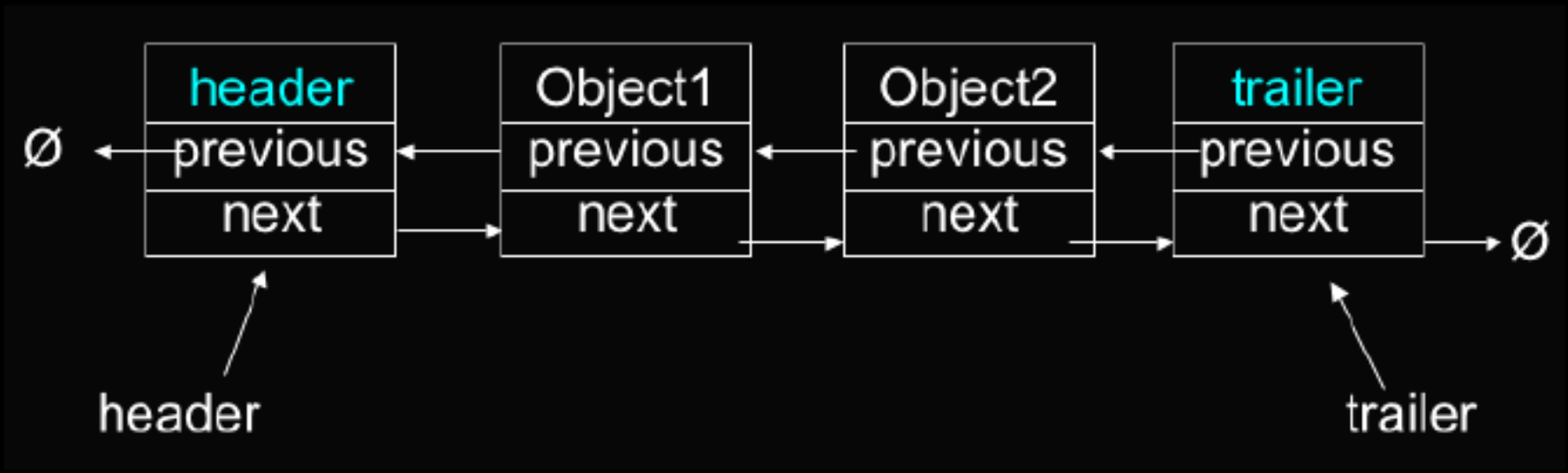
Taken from [Laf98].

Complexity analysis

	insertFirst	insert(i)	deleteFirst	delete(i)	get(i)
ArrayList	O(n)	O(n)	O(n)	O(n)	O(1)
LinkedList	O(1)	O(n)	O(1)	O(n)	O(n)

Table: Complexity analysis of the operations of LinkedList and ArrayList.

Implementation of Doubly-linked Lists



For more details, please check

- Robert Lafore. Data Structures and Algorithms in Java. Chapter 5.

Quiz questions

- Methods for both Linked List and Array List are the same, but implementations are different.
- Linked List is more efficient for insertion and ArrayList is more efficient for random access.

Quiz questions

- Methods for both Linked List and Array List are the same, but implementations are different.
- Linked List is more efficient for insertion and ArrayList is more efficient for random access.

References

- Please learn how to reference images, trademarks, videos and fragments of code.
- Avoid plagiarism

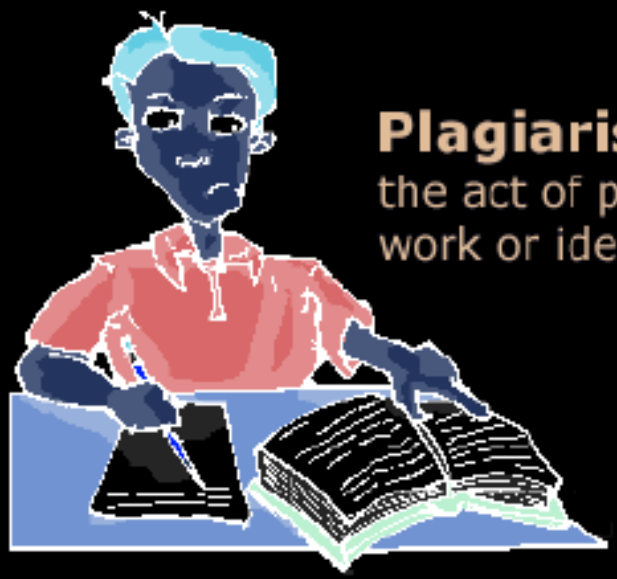


Figure: Figure about plagiarism, University of Malta [Uni09]

References

Robert Lafore.

Data Structures and Algorithms in Java.

QUE; 1 edition (8 November 2002), 1998.

University of Malta.

Plagarism — The act of presenting another's work or ideas as your own, 2009.

[Online; accessed 29-November-2013].

Further reading

- Linked Lists
  - Robert Lafore. Data Structures and Algorithms in Java. Chapter 3.